Stockholm Render - Project summary

DH2323: Computer Graphics with Interaction

Dmitrij Lioubartsev - dmitrijl@kth.se
Kristoffer Hallqvist - khallq@kth.se

Project description

The goal was to create a 3D rendering of the center of Stockholm, specifically of the island Stadsholmen (Gamla Stan) and perhaps its surroundings, using real-world data. It was not specified in more detail. The project has been about experimenting with different methods and finding something that works. The model rendering is done in Unity.

Work process

The project was worked on on-and-off from August 2014 to late May 2015. The work process included a lot of experimentation with various height data, libraries and programs, until finally finding something that works and works well. The final method can be (roughly) summarized as

1. Acquire an accurate height data point cloud from Lantmäteriet.
2. Parse the acquired point cloud, and downsample it.
3. Input the point cloud to the *Triangle* program (not made by us) to generate triangles for a mesh.
4. Assign the points and triangles to a mesh model in Unity.
5. Do some post-processing for classification and coloring.

A complete description of the every step of the project, in chronological order, can be found on the project blog, which serves as our main report. It describes how the project unfolded, which methods were tried and how the chosen solution works.

Links

- **Blog:** https://stockholmrender.wordpress.com/
- **Gihub, complete code base, Unity project and some images:** https://github.com/dmitrijl/dgi14_project
- **Triangle program:** https://www.cs.cmu.edu/~quake/triangle.html
- **Height data:** https://maps.slu.se/get/

Task allocation

The main contributors to this project are the authors (Kristoffer & Dmitrij). The parts not done by the authors include some shader code snippets, scanning and data gathering, animated water creation, as well as triangle construction. This has been properly credited and mentioned in the blog posts, where appropriate. The work distribution within the group is reflected in the blog. The author of each blog post is the main responsible, and the one who did most of the work on that particular area. Here is also a summary:

Dmitrij:

- All initial work with the 2D maps and SRTM height maps.
- Camera script
- Painful experimentation with PCL, libLAS, laspy, .las files, Triangle
- Design of chosen method, implementation of python program
- Design (and part implementation) of voxelgrid algorithm
- Getting the initial meshes rendered into Unity - C# script implementation
- Stitching together four meshes
- Adding basic water

Kristoffer:

- Cropping (isolating Stadsholmen)
- Model coloring (**not** writing the actual shaders)
- Simple and advanced building detection
- Spike removal
- Applying animated water to the model (**not** creating the water itself)
- Interactive controls for real-time environmental changes
- User HUD for viewing model parameters
- Web player demo creation and uploading

Further development possibilities

There are a few ways in which this project could be developed further, some of which are mentioned on the project blog. One thing you could do is to develop some more advanced form of building detection. You could develop smarter algorithms or even apply them to the point cloud before it is downsampled, to improve the accuracy. These areas can then be flattened to make place for more advanced building models, which can be made as a separate project and integrated later on.

Another possibility is to apply road detection, which would allow for roads to be implemented in the model. These can then be filled with moving cars or pedestrians to make the model more lifelike, again probably as a separate project for later integration.

The data is also quite heavily downsampled - from millions of points to tens of thousands. So it is possible to achieve even higher resolutions, although that would require more work to smoothen outliers.